

## Eisdealer-Game

Die Sommer werden immer heißer, das treibt unschuldige Leute in die gierigen Arme der gierigen Eisdealer. Wir wollen wenigstens mit einem kleinen Spielchen von diesem Effekt profitieren. Konzipiere und produziere eine Anwendung, mit der Nutzer:in sich in die Rolle des Betreibers einer Eisdiele versetzen können.

In einer kleinen Simulation kommen dann Kund:in sichtbar zur Eisdiele und setzen sich an die freien Plätze oder werden vom Nutzer:in dorthin gebracht, wo sie dann Eisbecher mit unterschiedlichen Anzahlen von Eiskugeln verschiedener Sorten ggf. mit Saucen und Toppings bestellen. Dann ist es **die Aufgabe des Nutzer:in**, diese **Kund:in mit den bestellten Eisbechern zu versorgen**, diese also **interaktiv zusammenzustellen** und **auszugeben**. Die Kund:in fressen schließlich langsam ihre Eisbecher, verlangen die **Quittung**, worauf das Nutzer:in die Zahlung entgegen nimmt, und verschwinden wieder. Ist die Eisdiele zu voll, müssen Kund:in draußen warten.

Die Kund:in werden so dargestellt, dass man ihre Stimmung gut erkennen kann. Diese sinkt, wenn sie lange warten müssen, ob innerhalb oder außerhalb der Eisdiele, den falschen Eisbecher erhalten oder die Preise zu hoch sind. Sie steigt, wenn sie schnell bedient werden. Fortlaufend werden die aktuellen Einnahmen seit Beginn der Simulation dargestellt.

Die angebotenen Eissorten, Toppings und Saucen sind im Code in einer passenden Datenstruktur vorgehalten und werden dargestellt, so dass das Nutzer:in damit interagieren kann. Es muss also möglich sein, **das Angebot** zu **ändern** nur **durch** Änderung dieser Datenstruktur, ohne weitere Änderungen im Code vornehmen zu müssen.

### Aufgaben S24

[L00\\_Preparation](#)  
[L01\\_Zufallsgedicht](#)  
[Übung: L01\\_Boxes](#)  
[L02\\_EventInspector](#)  
[L03\\_Einkaufsliste\\_Formular](#)  
[L04\\_Einkaufsliste\\_Datenstruktur](#)  
[L08.1\\_GenerativeKunst](#)  
[L09\\_EntenteichClasses](#)  
[L10\\_Ententeich: Polymorphie](#)  
[L11\\_Ententeich: Interaktiv](#)  
[Abschlussarbeit: EisdealerLtd](#)

Clone this wiki locally

<https://github.com/JirkaDellOro/>

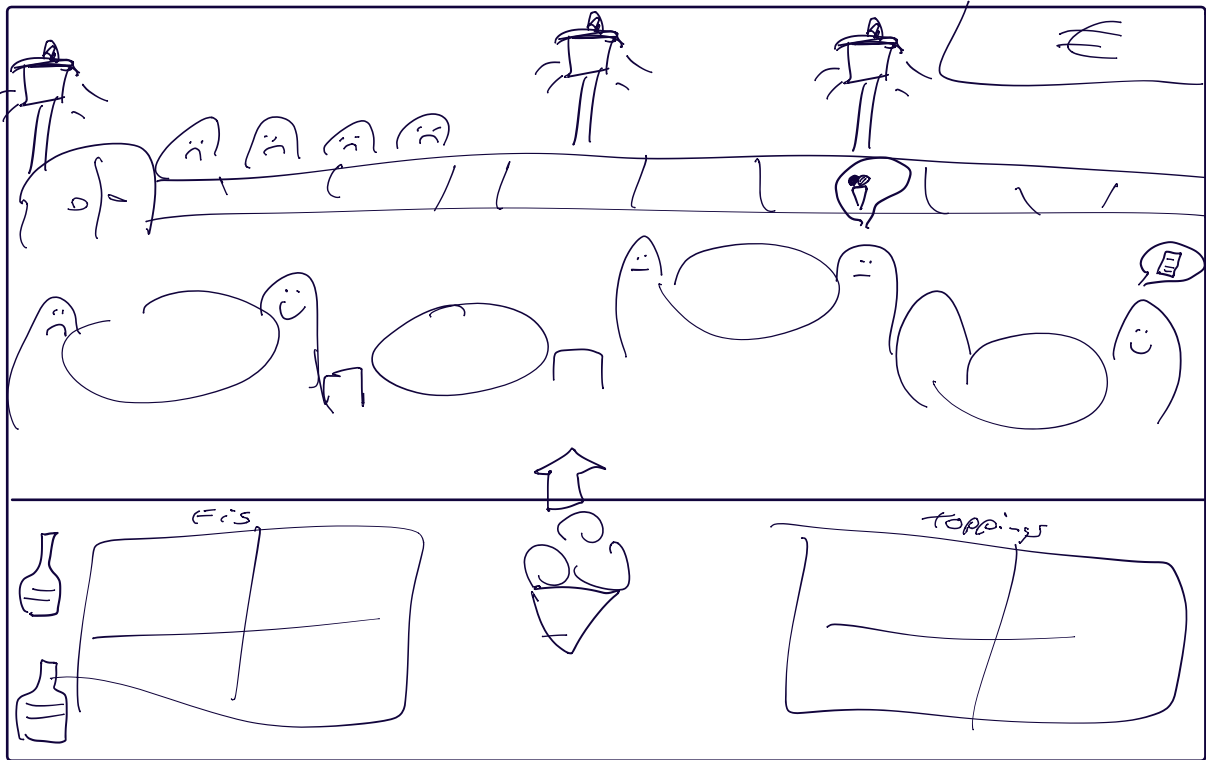
### Hinweise und Tipps

- es genügt völlig, die Eisdeale in der Aufsicht darzustellen.
- es genügt, die Akteure als Smileys darzustellen, an deren Ausdruck die Stimmung abgelesen werden kann.
- es sollen ein oder mehrere Arrays genutzt werden, um die Entitäten zu verwalten.

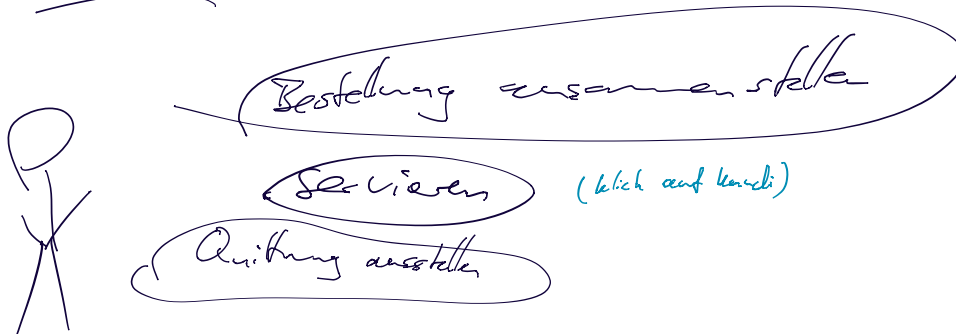
→ iron. dabei  
 Gewicht Interaktion  
 von Nutzer:in

PC

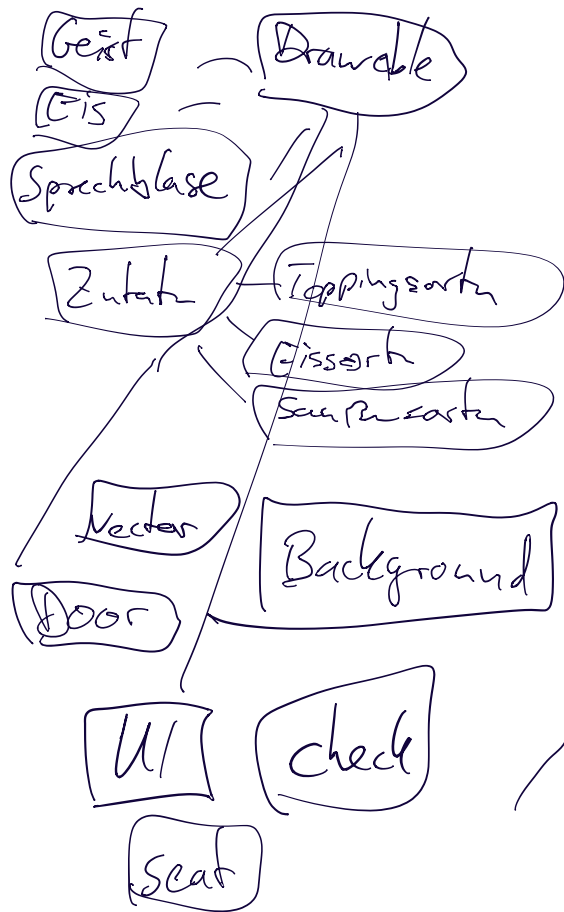
eine von langer nicht  
 alles schwarz



use-case



# Klassen



Geist

want: string  
emotion: string  
scaled: boolean  
move()  
order()  
changeEmotion()

Sprechblase

content: (Klasse) Eis / string

Zutaten

type: string

Eis

scale: size

Drawable

Position: Vector  
draw()

UI

interaction()

Door

state: boolean  
change()

Hintergrund: Mauern, Laternen, Himmel, Boden, Tische, Theke

Interaktion: Geister, Socke, Toppings, Sauce, Cafeteria

Toppings user  $\rightarrow$  if U/  $\rightarrow$  draw  $\square$   
if not  $\rightarrow$  draw  $\odot$

Wie auf Seat zugeifern aus anderen  
Klassen  $\rightarrow$  seat.free == true

if ghost search  $\rightarrow$  order()

if ghost has ICE  $\rightarrow$  3-10 sec when  
done  $\rightarrow$  check

if seat of seats[] empty  $\rightarrow$  door open

if door open  $\rightarrow$  ghost take seats